UNITED STATES PATENT APPLICATION

for

A USER INTERFACE FOR EFFICIENTLY BROWSING AN ELECTRONIC DOCUMENT USING DATA-DRIVEN TABS

Inventor:

Manfred Bartz

Marat Zhaksilikov

Doug Anderson

prepared by:

WAGNER, MURABITO & HAO LLP

Two North Market Street

Third Floor

San Jose, CA 95113

(408) 938-9060

A USER INTERFACE FOR EFFICIENTLY BROWSING AN ELECTRONIC DOCUMENT USING DATA-DRIVEN TABS

FIELD OF INVENTION

The present invention generally relates to software applications. Specifically, the present invention relates to a design utility within an application, in particular an application used for designing a microcontroller.

BACKGROUND OF THE INVENTION

As computer systems have proliferated throughout society, and network and Internet access continues to expand, users are confronted with an increasing amount of electronic documents. Electronic documents can contain information on any subject, from newspaper articles to television listings. Due to the essentially infinite amount of information available to be included in electronic form, the size of an electronic document can vary greatly.

Some electronic documents are small, only requiring a modest amount of time to read. If a user is looking for a particular section of an electronic document, browsing through the document also requires only a modest amount of time. However, as the length of the document increases, reading and browsing time likewise increases. For documents that are the equivalent of several written pages (e.g., ten or more written pages), reading and browsing can be unwieldy and inefficient.

Unfortunately, for medium to large documents, current browsing techniques do not provide efficient tools for scanning and browsing. Furthermore, if the document is only viewed in a small portion of the screen, due to screen space needs of other applications or portions of the same application, the viewing of medium and large documents is difficult.

One example of a medium or large sized electronic document is a datasheet for use in designing, configuring and programming electronic devices. A datasheet is a document that provides technical details for peripherals used with a microprocessor. Depending on the complexity of the peripheral, a datasheet can vary from tens of pages to thousands of pages.

One conventional technique used in electronic document viewing is the predetermined placing of embedded anchors throughout the document at the beginning of sections of the document. Several document formats, in particular HTML and XML, provide for supplying embedded anchors throughout an electronic document. A listing of the anchors is presented at the beginning of the document. By interacting with one of the anchors, the browsing application scrolls the document to the corresponding anchor. However, if a user desires to go to a second section of the document, the user must either scroll back to the top of the document to where the listing of anchors is located or browse the document manually. Requiring a user to go back to the beginning of a

document is time consuming, and can be frustrating. Furthermore, for use in viewing datasheets, this technique can be particularly inefficient. Microcontroller designers typically must read several sections of several datasheets in order to select the optimal peripheral. Requiring a user to scroll to the top of a document several times per document is very ineffective in providing a microcontroller designer with the tools needed to select peripherals.

Another conventional technique used in electronic document viewing is to provide a contents window adjacent to the electronic document. The contents window has links to embedded anchors throughout the electronic document. By interacting with a link, the browsing application scrolls the document to the appropriate anchor. However, a contents window requires a substantial amount of screen space. In a number of applications requiring the use of medium or large electronic documents, screen space is at a premium, and allocating a portion of the screen space for a contents window is undesirable. Particularly, in design tools requiring the viewing of multiple windows, available screen space is at a minimum.

SUMMARY OF THE INVENTION

Accordingly, a need exists for a method or system for helping a user efficiently browse an electronic document. A need also exists for a method or system that satisfies the above need and does not require a user to return to the beginning of the electronic document to utilize the technique for efficient browsing. A need also exists for a method or system that satisfies the above need and consumes minimal screen space. A need also exists for a method or system that satisfies the above need and is easy to use.

The present invention provides a method and system for helping a user efficiently browse an electronic document using data-driven tabs. In one embodiment, the electronic document is a datasheet providing technical details of a corresponding user module for use in microcontroller design. A user module is a pre-configured circuit design operating on a microcontroller. In one embodiment, the microcontroller utilizes a programmable system on chip architecture.

According to the invention, a document (e.g., a datasheet) is scanned for indicators (e.g., embedded anchors). The indicators are for indicating a predetermined location within the document. The indicators may correspond to section headers, subject dividers or any other predetermined location within a document.

The datasheet description is read and graphic elements (e.g., tabs) are automatically rendered for each corresponding indicator, wherein a graphic element is rendered according to information within the indicator. In one embodiment, the indicator comprises a section title. Upon rendering the graphic element, the graphic element comprises the section title. The indicators are predetermined and are placed by the author of the document to aid in the efficient browsing of the document.

Interacting with one of the graphic elements allows a user to jump to a predetermined location within the datasheet. The graphic elements are operable for efficient navigation of the datasheet, allowing for a large datasheet to be easily viewed in a small area of a display.

In one embodiment, the document is an HTML document. In one embodiment, the indicators are embedded HTML anchors. In another embodiment, the document is an XML document.

In one embodiment, the document is selected from a catalog of documents. In another embodiment, wherein the document is a datasheet, the datasheet is selected by selecting a particular user module. In one embodiment, the user module is selected from a catalog of user modules.

In one embodiment, the graphic elements are rendered adjacent to said document. In one embodiment, a user interacting with a scroll bar for scrolling through said document activates a graphic element upon passing a corresponding indicator of said graphic element, such that a current location on said document is rendered.

These and other objects and advantages of the present invention will become obvious to those of ordinary skill in the art after having read the following detailed description of the preferred embodiments that are illustrated in the various drawing figures.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and form a part of this specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention:

Figure 1 is a block diagram of an exemplary computer system upon which embodiments of the present invention may be practiced.

Figure 2A is a block diagram of an exemplary programmable system on a chip (SoC) architecture used with one embodiment of the present invention.

Figure 2B is a block diagram of an exemplary arrangement of SoC blocks used with one embodiment of the present invention.

Figure 3 is a flowchart of a process used by a microcontroller design tool in accordance with one embodiment of the present invention.

Figure 4 illustrates an exemplary graphical user interface of an electronic document with data-driven tabs in accordance with one embodiment of the present invention.

Figure 5 illustrates an exemplary graphical user interface of a microcontroller design application having a datasheet with data-driven tabs in accordance with one embodiment of the present invention.

Figures 6A, 6B and 6C are exemplary screen shots of a microcontroller design application having a datasheet with data-driven tabs in accordance with one embodiment of the present invention.

Figure 7 is a block diagram of an exemplary tab generator for helping a user browse a document in accordance with one embodiment of the present invention.

Figure 8A is an illustration of an exemplary input electronic document with embedded anchors for generating tabs therefrom in accordance with one embodiment of the present invention.

Figure 8B is exemplary HTML code for an HTML document with embedded anchors for generating tabs therefrom in accordance with one embodiment of the present invention.

Figure 9 is a flowchart of a process for helping a user design a microcontroller in accordance with one embodiment of the present invention.

Figure 10 is a flowchart of a process for helping a user browse a document in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION

Reference will now be made in detail to the preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. While the invention will be described in conjunction with the preferred embodiments, it will be understood that they are not intended to limit the invention to these embodiments. On the contrary, the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention as defined by the appended claims. Furthermore, in the following detailed description of the present invention, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be obvious to one of ordinary skill in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, and circuits have not been described in detail so as not to unnecessarily obscure aspects of the present invention.

Some portions of the detailed descriptions that follow are presented in terms of procedures, logic blocks, processing, and other symbolic representations of operations on data bits within a computer memory. These descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. In the present application, a procedure, logic block, process, or the like, is conceived to be a self-consistent sequence of steps or

instructions leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, although not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated in a computer system. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as transactions, bits, values, elements, symbols, characters, fragments, pixels, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussions, it is appreciated that throughout the present invention, discussions utilizing terms such as "rendering," "receiving," "generating," "displaying," "scanning," "indicating," "interacting," "scrolling" or the like, refer to actions and processes (e.g., processes 300, 900 and 1000 of Figures 3, 9 and 10, respectively) of a computer system or similar electronic computing device. The computer system or similar electronic computing device manipulates and transforms data represented as physical (electronic) quantities within the computer system memories, registers or other such information storage, transmission or display devices. The present invention is well suited to use with other computer systems.

Refer now to Figure 1, which illustrates an exemplary computer system 190 upon which embodiments of the present invention may be practiced. In general, computer system 190 comprises bus 100 for communicating information, processor 101 coupled with bus 100 for processing information and instructions, random access (volatile) memory (RAM) 102 coupled with bus 100 for storing information and instructions for processor 101, read-only (non-volatile) memory (ROM) 103 coupled with bus 100 for storing static information and instructions for processor 101, data storage device 104 such as a magnetic or optical disk and disk drive coupled with bus 100 for storing information and instructions, an optional user output device such as display device 105 coupled to bus 100 for displaying information to the computer user, an optional user input device such as alphanumeric input device 106 including alphanumeric and function keys coupled to bus 100 for communicating information and command selections to processor 101, and an optional user input device such as cursor control device 107 coupled to bus 100 for communicating user input information and command selections to processor 101. Furthermore, an optional input/output (I/O) device 108 is used to couple computer system 190 onto, for example, a network.

Display device 105 utilized with computer system 190 may be a liquid crystal device, cathode ray tube, or other display device suitable for creating graphic images and alphanumeric characters recognizable to the user. Cursor control device 107 allows the computer user to dynamically signal the two-

dimensional movement of a visible symbol (pointer) on a display screen of display device 105. Many implementations of the cursor control device are known in the art including a trackball, mouse, joystick or special keys on alphanumeric input device 106 capable of signaling movement of a given direction or manner of displacement. It is to be appreciated that the cursor control 107 also may be directed and/or activated via input from the keyboard using special keys and key sequence commands. Alternatively, the cursor may be directed and/or activated via input from a number of specially adapted cursor directing devices.

The present invention is described in the context of a software tool, portions of which are comprised of computer-readable and computer-executable instructions which reside, for example, in computer-usable media of a computer system such as that exemplified by Figure 1. The present invention is primarily described as being used with a tool for designing configuring, programming, compiling, building (assembling), emulating, and debugging an embedded microcontroller, in particular a class of microcontrollers that provide analog and/or digital subsystems comprising many dynamically configurable blocks. An example of this class is referred to herein as a programmable system on a chip (PSoC). Additional information regarding PSoCs is provided in the co-pending, commonly-owned US Patent Application, Attorney Docket No. CYPR-CD00232, Serial No. _____, filed

October 22, 2001, by W. Snyder, and entitled "Microcontroller Programmable System on a Chip," hereby incorporated by reference.

Figure 2A is a block diagram of an integrated circuit (or microcontroller) 210 that exemplifies a microcontroller which uses the PSoC architecture. In the illustrated embodiment, integrated circuit 210 includes a system bus 211, and coupled to bus 211 are synchronous random access memory (SRAM) 212 for storing volatile or temporary data during firmware execution, central processing unit (CPU) 214 for processing information and instructions, flash read-only memory (ROM) 216 for holding instructions (e.g., firmware), input/output (I/O) pins 218 providing an interface with external devices and the like, and system on a chip (SoC) blocks 225. The SoC blocks 225 include analog blocks and digital blocks, which are further described below (see Figure 2B).

Referring to Figure 2B, an embodiment of SoC block 225 is depicted in greater detail. In this embodiment, SoC block 225 includes an analog functional block 230, a digital functional block 240, and a programmable interconnect 250. Analog block 220 includes, in the present embodiment, a matrix of interconnected analog blocks A1 through AN. The number N may be any number of analog blocks. Likewise, digital block 240 includes, in the present embodiment, a matrix of interconnected digital blocks D1 through DM. The number M may be any number of digital blocks. The analog blocks A1 through AN and the digital blocks D1 through DM are fundamental building

blocks that may be combined in different ways to accomplish different functions. Importantly, different combinations of blocks, producing different functions, may exist at different times within the same system. For example, a set of blocks configured to perform the function of analog-to-digital conversion may sample a signal. After processing that signal in the digital domain, those same blocks (perhaps in conjunction with a few others) may be recombined in a different configuration to perform the function of digital-to-analog conversion to produce an output signal.

Continuing with reference to Figure 2B, the internal matrices of analog blocks 230 and digital blocks 240 may be constituted, in one embodiment, by a routing matrix described further in the patent application referenced above. Analog blocks 230 and digital blocks 240 are electrically and/or communicatively coupled to programmable interconnect 250, in the present embodiment, by intra-block routing 235. Each individual functional unit, e.g., analog blocks A1 through AN and digital blocks D1 through DM, may communicate and interact with each and/or any other functional unit. Which functional unit communicates with which other functional unit is programmable, in the present embodiment, via the configurability of the programmable interconnect 250. The programmable interconnect 250 is connected via an internal input/output (I/O) bus 236 to pin-by-pin configurable I/O transceivers (pins) 218 (Figure 2A), which effect communicative coupling between integrated circuit 210 (Figure 2A) and external modalities. The total pin count of

pin-by-pin configurable I/O pins 218 may vary from one application to another, depending on the system device under consideration.

With reference next to Figure 3, process 300 illustrates exemplary steps used by a microcontroller design tool in accordance with one embodiment of the present invention. The purpose of process 300 is to configure, program, compile, build, emulate and debug a customized microcontroller (a "target device") based on the integrated circuit 210 and SoC blocks 225 of Figures 2A and 2B.

In one embodiment, process 300 of Figure 3 is carried out by a processor under the control of computer-readable and computer-executable instructions. The computer-readable and computer-executable instructions reside, for example, in data storage features such as computer usable volatile memory 102, computer-usable non-volatile memory 103, and/or data storage device 104 of Figure 1. The computer-readable and computer-executable instructions are used to control or operate in conjunction with, for example, central processing unit 101 of Figure 1.

Although specific steps are disclosed in process 300 of Figure 3, such steps are exemplary. That is, the present invention is well suited to use with various other steps or variations of the steps recited in process 300. Additionally, for purposes of clarity and brevity, the following discussion and

examples specifically deal with a microcontroller design tool. The present invention, however, is not limited solely to use with a microcontroller design tool. Instead, the present invention is well suited to use with other types of computer-aided hardware and software design systems in which it is necessary to accomplish a multitude of tasks as part of an overall process.

In step 310, applicable "user modules" are selected. A user module, as used herein, is a preconfigured function that may be based on more than one SoC blocks. A user module, once placed and programmed, will work as a peripheral on the target device. At any time in process 300, user modules may be added to or removed from the target device.

The selected user modules can then "placed" or "mapped" onto the SoC blocks 225 of Figure 2B. Once a user module is placed, its parameters can be viewed and modified as needed. Global parameters used by all of the user modules (for example, CPU clock speed) can also be set.

Continuing with step 310 of Figure 3, interconnections between the selected user modules can be specified, either as each user module is placed or afterwards. The pin-out for each PSoC block can be specified, making a connection between the software configuration and the hardware of the target device.

In step 320, application files can be generated. When application files are generated, existing assembly-source and C compiler code are updated for all device configurations, and application program interfaces (APIs) and interrupt service routines (ISRs) are generated.

In step 330, the desired functionality can be programmed into the target device. Source code files can be edited, added or removed.

In step 340, the assembler process can be executed. The assembler operates on an assembly-language source to produce executable code. This code is compiled and built into an executable file that can be downloaded into an emulator, where the functionality of the target device can be emulated and debugged.

In step 350, the target device can be "built." Building the target device links all the programmed functionalities of the source files (including device configuration), which are downloaded to a file for debugging.

In step 360, the target device can be emulated using an in-circuit emulator for debugging. The emulator allows the target device to be tested in a hardware environment while device activity is viewed and debugged in a software environment.

Figure 4 illustrates an exemplary graphical user interface (GUI) of an electronic document with data-driven tabs that is generated in accordance with one embodiment of the present invention. In one embodiment, these GUIs are displayed on display device 105 of computer system 190 (Figure 1). It is appreciated that these GUIs are exemplary only, and that they may include different numbers and shapes of elements and windows other than those that are illustrated.

Figure 4 shows a GUI 400 comprising electronic document 410, graphic elements 420a-420h, scroll bar 430 and scroll box 440. In one embodiment, the graphic elements are rendered as tabs. As will be seen, the graphic elements correspond to locations within electronic document 410 in order to facilitate easy and efficient browsing of the electronic document.

In accordance with the present invention, each graphic element corresponds to a location within electronic document 410. A graphic element, when selected (e.g., interacted with) jumps or scrolls the electronic document to the corresponding location within the document which the browser displayed in the screen. In one embodiment, highlighting or changing the color of the graphic element indicates a selected graphic element. In another embodiment, a selected graphic element is by bolding or otherwise altering the text within the graphic element. In another embodiment, a selected graphic

element is indicated by placing the selected graphic element in the foreground of all graphic elements (e.g., graphic element 420d of Figure 4).

Elements are selected by a user using well-known GUI techniques. That is, for example, a user can position a cursor over an element and "click" a cursor control element (e.g., a mouse) to select an element. When the elements are rendered as tabs, a selected element is depicted as described above. In general, a selected element is rendered in a way that allows the user to readily determine which element has been selected.

The graphic elements are determined by the contents of electronic document 410. Electronic document 410 comprises a number of indicators for indicating a predetermined location within the document. In one embodiment, the indicators are embedded anchors. It should be appreciated that the indicators are predefined by the author of electronic document 410. For example, consider the case where electronic document 410 is a novel divided into chapters. The author may desire to place indicators at the beginning of each chapter, such that graphic elements are rendered for each chapter of the novel, allowing a user to jump to particular chapters.

In one embodiment, electronic document 410 is a datasheet providing technical details of a corresponding user module (e.g., peripheral), wherein a user module is a pre-configured function operating on a microcontroller. For

example, indicators placed in the current embodiment would correspond to particular sections of the datasheet (e.g., Overview, Diagram, Features and Registers). It should be appreciated that the specific indicators would be determined by the particular datasheet selected, and that the above indicators are exemplary only.

In one embodiment, the electronic document is an HTML document. It should be appreciated that any format for generating or creating electronic documents may be used (e.g., XML), and that the present invention is not intended to be limited to any particular form of electronic document. In one embodiment, the indicators are embedded HTML anchors within the document. It should be appreciated that the indicators are typically hidden from a user, but that the content of the indicator is typically visible.

It should be appreciated that an electronic document can have any number of associated graphic elements, and is not limited to the eight graphic elements 420a-420h as shown in Figure 4.

In one embodiment, when a graphic element is selected, electronic document 410 jumps to the location of the corresponding graphic element. In one embodiment, when a graphic element is selected, scroll box 440 moves to a corresponding location of scroll bar 430, indicating a current location within electronic document 410. Likewise, in one embodiment, when scroll box 440

is moved to a different location within scroll bar 430, the graphic element corresponding to the location within electronic document 440 is selected.

Figure 5 illustrates an exemplary workspace 500 as displayed on a graphical user interface (GUI) of a microcontroller design application having a datasheet with data-driven tabs in accordance with one embodiment of the present invention. In one embodiment, these GUIs are displayed on display device 105 of computer system 190 (Figure 1). It is appreciated that these GUIs are exemplary only, and that they may include different numbers and shapes of elements and windows other than those that are illustrated.

Figure 5 shows a workspace 500 comprising a number of open windows 510, 520, 530 and 540, a datasheet 550, a number of graphic elements 560a-560h, scroll bar 570 and scroll box 580. A datasheet provides specific technical details for a user module, such as timing diagrams, pin outs, stacks, a detailed description and other information essential to microcontroller designers for selecting the correct user module. In one embodiment, the graphic elements are rendered as tabs. As will be seen, the graphic elements correspond to locations within datasheet 550 in order to facilitate easy and efficient browsing of datasheet 550.

In one embodiment, window 510 is a catalog of available user modules (e.g., peripheral), wherein a user module is a pre-configured function operating

on a microcontroller. In one embodiment, window 520 is a listing of selected user modules. In one embodiment, window 530 is a schematic for a selected user module. In one embodiment, window 540 is a histogram showing the available resources of a microcontroller.

In accordance with the present invention, each graphic element corresponds to a location within datasheet 550. A graphic element, when selected (e.g., interacted with) jumps or scrolls the electronic document to the corresponding location within the document. In one embodiment, highlighting or changing the color of the graphic element indicates a selected graphic element. In another embodiment, a selected graphic element is by bolding or otherwise altering the text within the graphic element. In another embodiment, a selected graphic element is indicated by placing the selected graphic element in the foreground of all graphic elements (e.g., graphic element 560d of Figure 5).

As described above, elements are selected by a user in a conventional manner. That is, for example, a user can position a cursor over an element and "click" a cursor control element (e.g., a mouse) to select an element. When the elements are rendered as tabs, a selected element is depicted as described above. In general, a selected element is rendered in a way that allows the user to readily determine which element has been selected.

The graphic elements are determined by the contents of datasheet 550. Datasheet 550 comprises a number of indicators for indicating a predetermined location within the document. In one embodiment, the indicators are embedded anchors. It should be appreciated that the indicators are predefined by the author of datasheet 550. For example, indicators placed in the current embodiment would correspond to particular sections of the datasheet (e.g., Overview, Diagram, Features and Registers). It should be appreciated that the specific indicators would be determined by the particular datasheet selected, and that the above indicators are exemplary only.

In one embodiment, datasheet 550 is an HTML document. It should be appreciated that any format for generating or creating electronic documents may be used (e.g., XML), and that the present invention is not intended to be limited to any particular form of electronic document. In one embodiment, the indicators are embedded HTML anchors within datasheet 550. It should be appreciated that the indicators are typically hidden from a user, but that the content of the indicator is typically visible.

It should be appreciated that an electronic document can have any number of associated graphic elements, and is not limited to the eight graphic elements 560a-560h as shown in Figure 5.

In one embodiment, when a graphic element is selected, datasheet 550 jumps to the location of the corresponding graphic element. In one embodiment, when a graphic element is selected, scroll box 580 moves to a corresponding location of scroll bar 570, indicating a current location within datasheet 550. Likewise, in one embodiment, when scroll box 580 is moved to a different location within scroll bar 570, the graphic element corresponding to the location within datasheet 550 is selected.

Figures 6A, 6B and 6C are exemplary screen shots of a microcontroller design application having a datasheet with data-driven tabs in accordance with one embodiment of the present invention. Workspaces 600, 601 and 602 of Figures 6A, 6B and 6C, respectively, comprise catalog window 605 comprising a catalog of all available user modules, selected user module window 610 comprising a listing of all user modules selected from catalog window 605, selected user module 615, schematic window 620 illustrating a schematic of selected user module 615, histogram window 625 illustrating available resources and datasheet 630. Datasheet 630 comprises data-driven tabs 640, scroll bar 645 and scroll box 650.

Figure 6A illustrates an example where tab 635 of datasheet 630 is selected. Accordingly, datasheet 630 is scrolled to the location of the content associated with tab 635. Likewise, scroll box 650 is located at the appropriate location within scroll bar 645. In the present embodiment, tab 635 is entitled

"Resources." Thus, the resources of selected user module 615 as described in datasheet 630.

Figure 6B illustrates an example where tab 636 of datasheet 630 is selected. Accordingly, datasheet 630 is scrolled to the location of the content associated with tab 636. Likewise, scroll box 650 is located at the appropriate location within scroll bar 645. In the present embodiment, tab 636 is entitled "Features." Thus, the features and benefits of selected user module 615 as described in datasheet 630.

Similarly, Figure 6C illustrates an example where tab 637 of datasheet 630 is selected. Accordingly, datasheet 630 is scrolled to the location of the content associated with tab 637. Likewise, scroll box 650 is located at the appropriate location within scroll bar 645. In the present embodiment, tab 637 is entitled "Placement." Thus, the placement of selected user module 615 as described in datasheet 630.

Figure 7 is a block diagram of an exemplary tab generator 700 for helping a user browse an electronic document 710 in accordance with one embodiment of the present invention. In one embodiment, tab generator 700 is implemented in software running on a computer system (e.g., computer system 190 of Figure 1). In another embodiment, tab generator 700 is

implemented in hardware. In another embodiment, tab generator 700 is implemented in firmware.

Display engine 720 receives an input of electronic document 710. In one embodiment, electronic document 710 is an HTML document. It should be appreciated that any format for generating or creating electronic documents may be used (e.g., XML), and that the present invention is not intended to be limited to any particular form of electronic document. In one embodiment, electronic document 710 comprises indicators or embedded anchors. In one embodiment, the indicators are embedded HTML anchors. It should be appreciated that the indicators are typically hidden from a user, but that the content of the indicator is typically visible.

It should be appreciated that electronic document is created by an author who determines the characteristics of the document. The author selects a format for generating the document, determines the content of the document, and is responsible for the appearance of the document. In one embodiment, the author includes indicators or embedded anchors in the document to designate separate sections or locations of the document.

With reference now to Figure 8A, an illustration of an exemplary input electronic document 710 is shown. In one embodiment, electronic document 710 comprises a number of sections 810, each section having an embedded

anchors 820 for generating tabs therefrom in accordance with one embodiment of the present invention. As explained above, the embedded anchors are placed by the author of electronic document 710, and are used to help a user efficiently browse the document.

With reference now to Figure 8B, exemplary HTML code for an HTML document 850 with embedded anchors 860 and 870 for generating tabs therefrom is shown. In the one embodiment, HTML document 850 is a datasheet for a user module in accordance with the present invention. In the present embodiment, embedded anchors 860 and 870 are indicated in HTML code as "tab_ADCINC12" and "tab_Resources," respectively.

Returning to Figure 7, display engine 720, upon receiving electronic document 710, scans electronic document 710 for indicators or embedded anchors. Display engine 720 generates tabs (e.g., graphic elements 420a-h of Figure 4) based on the indicators or embedded anchors. In one embodiment, display engine 720 performs a process for helping a user browse a document as described in process 1000 of Figure 10.

The data contained within electronic document 710 is used to generate the tabs. As the tabs are generated based on the data within each electronic document, and not based on the application itself, the content of the tabs is

data-driven. Upon generating the tabs, GUI 730 displays electronic document 710 with corresponding tabs.

Figure 9 is a flowchart of a process 900 for helping a user design a microcontroller in accordance with one embodiment of the present invention and Figure 10 is a flowchart of a process 1000 for helping a user browse a document in accordance with one embodiment of the present invention. In one embodiment, processes 900 and 1000 are carried out by a processor under the control of computer-readable and computer-executable instructions. The computer-readable and computer-executable instructions reside, for example, in data storage features such as computer usable volatile memory 102, computer-usable non-volatile memory 103, and/or data storage device 104 of Figure 1. The computer-readable and computer-executable instructions are used to control or operate in conjunction with, for example, central processing unit 101 of Figure 1.

Although specific steps are disclosed in processes 900 and 1000, such steps are exemplary. That is, the present invention is well suited to use with various other steps or variations of the steps recited in processes 900 and 1000. Additionally, for purposes of clarity and brevity, the following discussion and examples specifically deal with a microcontroller design tool. The present invention, however, is not limited solely to use with a microcontroller design tool. Instead, the present invention is well suited to use with other types of

computer-aided hardware and software design systems in which it is necessary to accomplish a multitude of tasks as part of an overall process.

With reference first to process 900 of Figure 9, at step 910, a workspace is rendered in a GUI displayed on a display device of a computer system. In the present embodiment, the workspace has multiple windows, wherein one window comprises at least one selectable user module, wherein the user module is a pre-configured function operating with a microcontroller.

At step 920, input is received indicating a selection of a user module. In one embodiment, a user module is selected in response to user selection or interaction.

At step 930, a datasheet is automatically rendered in a second window of the workspace. A datasheet provides technical details corresponding to the selected user module. In this manner, datasheets are rendered within a microcontroller design application for aiding a user in designing a microcontroller.

Referring now to process 1000 of Figure 10, at step 1010, an electronic document is scanned for embedded anchors or indicators. The embedded anchors or indicators are for indicating a predetermined location within the

electronic document. It should be appreciated that the electronic document can be rendered in any format for creating electronic documents.

At step 1020, tabs (e.g., graphic elements) are generated based on the content of each embedded anchor. Each tab is rendered according to information within the indicator, and thus is data-driven. That is, the tabs rendered are based on the information within the electronic document itself, rather than the application for rendering the electronic document. The tab is for communicating information concerning the location of specific content within an electronic document. Interacting with one of the tabs allows a user to jump to a predetermined location within the electronic document corresponding to the tab. In this manner, data-drive tabs are generated for easy browsing of electronic documents.

In summary, the present invention is used to render a workspace for helping a user design a microcontroller, rendering a workspace comprising a datasheet for a selectable user module and providing easy navigation of the datasheet. The user is provided with a workspace for helping a user select the desirable user modules for use with a microcontroller, while directly providing the user with the technical information to select the appropriate user module.

The preferred embodiment of the present invention, a method of rendering a workspace for helping a user design a microcontroller, is thus

described. While the present invention has been described in particular embodiments, it should be appreciated that the present invention should not be construed as limited by such embodiments, but rather construed according to the below claims.